CS 271 - Project 0011

James Le

February 15, 2017

1. Where in a min-heap might the largest element reside, assuming that all elements are distinct?

   Since the parent is greater or equal to its children, the smallest element must be a leaf node.

2. Is an array in sorted order a min-heap? Why or why not?

   An array sorted from lowest to highest is a min-heap when using the array-based heap implementation. The heap property that the parent node is greater than its child nodes (2i + 1 and 2i + 2, using zero-based arrays) holds for all nodes that have children.

3. Implement a MinHeap template class, and the heap sort algorithm. The heap sort method should return a copy of the array in ascending sorted order.

4. Plot the running time of heap sort for a variety of sufficiently large input sizes and compare this to the running time of the other sorting algorithms that you implemented in CS 173.

   Heap Sort

   - Best Case: $\Omega(n \log(n))$
   - Average Case: $\Theta(n \log(n))$
   - Worst Case: $O(n \log(n))$

   Quick Sort

   - Best Case: $\Omega(n \log(n))$
   - Average Case: $\Theta(n \log(n))$
   - Worst Case: $O(n^2)$

   Merge Sort

   - Best Case: $\Omega(n \log(n))$
   - Average Case: $\Theta(n \log(n))$
   - Worst Case: $O(n \log(n))$

   Bubble Sort

   - Best Case: $\Omega(n)$
   - Average Case: $\Theta(n^2)$
   - Worst Case: $O(n^2)$

   Insertion Sort

   - Best Case: $\Omega(n)$
   - Average Case: $\Theta(n^2)$
   - Worst Case: $O(n^2)$

   Selection Sort

   - Best Case: $\Omega(n^2)$
   - Average Case: $\Theta(n^2)$
   - Worst Case: $O(n^2)$

5. What is the asymptotic time complexity of the heap sort algorithm on an array that is already sorted? What is the asymptotic time complexity on an array that is in reverse order? What is the best case asymptotic time complexity of heap sort, and on what kind of input does it occur?

   - On an array that is already sorted, the asymptotic time complexity of heap sort algorithm is $O(1)$.
   - On an array that is in reverse order (worst case), the asymptotic time complexity of heap sort algorithm is $O(n \log n)$.
   - The best case asymptotic time complexity of heap sort is $\Omega(n \log n)$.